# Fast Voronoi Diagrams and Offsets on Triangulated Surfaces

Ron Kimmel and James A. Sethian

**Abstract.** We apply the Fast Marching Method on triangulated domains to efficiently compute Voronoi diagrams and offset curves on triangulated manifolds. The computational complexity of the proposed algorithm is optimal, $O(M \log M)$, where $M$ is the number of vertices. The algorithm also applies to weighted domains in which a different cost is assigned to each surface point.

## §1. Introduction

Voronoi diagrams play important roles in many research fields such as robotic navigation and control, image processing, computer graphics, computational geometry, pattern recognition, and computer vision. Its Euclidean version, for which there is an efficient implementation, is a building block in many applications.

The Voronoi diagram sets boundaries between a given set of source points, and splits the domain into regions such that each region corresponds to the closest neighborhood of a source point from the given set. Let our domain be $D$, let the set of given $n$ points be $\{p_j \in D, j \in 0,..,n-1\}$, and the distance between two points $p, q \in D$ be $d(p,q)$. Then the Voronoi region $G_i$ corresponds to the set of points $p \in D$ such that $d(p,p_i) < d(p,p_j), \forall j \neq i$.

Offsets computation is often used in approximation and singularity theories, and comes into practice in computer aided design (CAD) and numerical control (NC machines). Given a curve and its embedding space, an offset curve is defined by a set of points with a given fixed distance from the original curve.

There are some numerical and topological difficulties, even in the computation of offsets for curves in the 2D Euclidean plane, e.g. the formation of singularities in the curvature, self intersection of the offsetting curve, and the fact that an offset of a polynomial parametrized curve is not necessarily polynomial. Some of the numerical difficulties were addressed in [9], where

Curve and Surface Design: Saint-Malo 1999
Pierre-Jean Laurent, Paul Sablonnière, and Larry L. Schumaker (eds.), pp. 193–202.

ISBN 0-8265-1356-5.

the Osher-Sethian level set method [16,20], which grew out of Sethian's earlier work on curve evolution, see [21], was used to overcome the topological changes.

Efficient construction of distance maps, minimal geodesics, Voronoi diagrams, and offset curves for non-flat and weighted domains is a challenging problem, see e.g. [15,13,8,12,6,10]. The core of our approach is Sethian's Fast Marching Method, [22,19,20] which solves the Eikonal equation on a rectangular orthogonal mesh in $O(M \log M)$ steps, where $M$ is the total number of grid points. Contingent upon the triangulated upwind and monotonic update schemes given by Barth and Sethian [1], this technique was extended to triangulated surfaces by Kimmel and Sethian in [11]. The triangulated version of the Fast Marching Method has the same computational complexity, and solves the Eikonal equation on triangulated domains in $O(M \log M)$ steps, where $M$ is the number of vertices. Using this technique, one can compute distances on curved manifolds with local weights. For other applications which rely on the Fast Marching Method, see [14,4].

Here we apply our method to compute Voronoi diagrams of a given set of points (or regions), and to find offsets from curves and points on triangulated manifolds. The computational complexity of the proposed algorithm is optimal $O(M \log M)$, its implementation is simple, and it also applies to weighted domains in which a different cost is assigned to each surface point.

The key idea is based on upwind finite difference operators as numerically consistent approximation to the differential operators in the Eikonal equation. Such an approximation selects the correct viscosity solution. The upwind operators allow us to construct a solution to the Eikonal equation by optimally sorting the updated points using a heap structure.

The outline of this paper is as follows. The key for fast computation of offsets and Voronoi diagrams is a fast algorithm for computing the distance. Hence, we first comment on the connection between the Eikonal equation and distance maps on weighted domains. We refer the reader to Sethian's Fast Marching Method for solving the Eikonal equation and for computing distance maps on orthogonal grids, and to [11] for details on our extension for computing the solution on triangulated domains. We then apply the method for the computation of fast Voronoi diagrams and offsets on triangulated manifolds.

## §2. Fast Marching Method and the Eikonal Equation

We first explore some aspects of distance computation on weighted domains. In order to compute the distance between two points, we need to define a measure of length. A definition of an arclength allows us to measure distance by integrating the arclength along a curve connecting two points. The distance between the points corresponds to the length of the shortest curve connecting them.

Given a 2D weighted flat domain, or in other words an isotropic nonhomogeneous domain, the distance may be defined via the arclength definition. For example, the arclength may be written as a function of the $x$ and $y$ Cartesian coordinates of the planar domain

$$ds^2 = \mathcal{F}(x,y)^2(dx^2 + dy^2),$$

where $\mathcal{F}(x,y) : R^2 \rightarrow R^+$ is a function that defines a weight for each point in the domain.

The distance map $T(x,y)$ from a given point $p_0$ assigns a scalar value to each point in the domain that corresponds to its distance from $p_0$. It is easy to show, see e.g. [2], that the gradient magnitude of the distance map is proportional to the weight function at each point

$$|\nabla T(x,y)| = \mathcal{F}(x,y),$$

where $|\nabla T| \equiv \sqrt{T_x^2 + T_y^2}$. This equation is known as the Eikonal equation. The 'viscosity' solution to the Eikonal equation coupled with the boundary condition $T(p_0) = 0$ results in the desired distance map.

Our first goal is to solve the Eikonal equation. The key is to construct a numerical approximation to the gradient magnitude that selects an appropriate 'weak solution'. Consider the following upwind approximation to the gradient, given by

$$\left(\max(D_{ij}^{-x}T, -D_{ij}^{+x}T, 0)^2 + \max(D_{ij}^{-y}T, -D_{ij}^{+y}T, 0)^2\right)^{1/2} = \mathcal{F}_{ij},$$

where for example $D_{ij}^{-x}T \equiv \frac{T_{ij} - T_{i-1,j}}{h}$ is the standard backwards derivative approximation, and $T_{ij} \equiv T(i\Delta x, j\Delta y)$. The use of upwind schemes in hyperbolic equations is well known, see for example, Godunov's paper from 1959 [7]. For Hamilton-Jacobi equations, see e.g. [17,3].

The solution $T$ can be systematically constructed in an upwind fashion. The upwind difference approximation of the above equation means that information propagates one way from smaller values of $T$ to larger values. The Fast Marching Method exploits this order of events. A point gets updated only by points with smaller values. This 'monotone property' allows us to keep a front of candidate points that tracks the flow of information, ordered in a heap tree structure in which the root is always the smallest value. An update of an element in the heap tree is done in $O(\log M)$ operations. Thereby, the total computational complexity is $O(M \log M)$. We refer to [22,19,20] for further details on the Fast Marching Method.

One could recognize similarity to Dijkstra's method [5,18] that computes minimum costs of paths on networks. Dijkstra algorithm would obviously fail to consistently solve our geometric problems. Actually, any graph-search-based algorithm induces the artificial metric imposed by the graph network, and would be inconsistent with the continuous case, and thus fail to converge as the graph resolution is refined.

The Fast Marching Method that works for orthogonal grids may be viewed as a selection for the update of one of the four right angle triangles that share the same vertex. The extension to triangulated domains is motivated by this observation, by the geometric interpretation of the update step, and by an additional special treatment of obtuse angles. We refer to [11] for details on the extension of the fast marching method to triangulated domains. It is also based on a finite difference approximation to the Eikonal equation, this time on the surface, monotone by construction, consistent, upwind, and converges to the viscosity solution.

## §3. Offsets and Voronoi Diagrams

We have an algorithm to compute distances on triangulated manifolds, and hence construct offset curves. First, we solve the Eikonal equation with speed $\mathcal{F} = 1$ on the triangulated surface to compute the distance from a source point or a region that defines an initial curve. We then find the equal geodesic distance curves on the surface by interpolating the intersection with a constant threshold using a 'marching triangle' procedure, again an $O(M)$ operation. The offsets on the triangulated surface, or the equi-geodesic-distance curves, are shown in Figure 1. The black curve is the original curve, and the white curves are the offsets.

Figure 2 presents Voronoi diagrams on several beads and a synthetic head. We first compute the distance from each of the initial given source points simultaneously using a single heap structure, and allow one vertex overlap between distance maps form different sources. The complexity for the distance computation is still $O(M \log M)$. Next, we 'march' along the triangles, and for each triangle linearly interpolate the intersection curve between the two different distance maps, again an $O(M)$ operation.

The algorithm complexity remains the same as we add weights to the surface. In Figures 3 and 4 a different cost is assigned to each vertex. The cost, or weight function, is texture mapped onto the triangulated surface. The weighted offsets, or weighted equal geodesic distance contours are shown in Figure 3, while weighted geodesic Voronoi diagrams for several surfaces are presented in Figure 4. In both examples, dark intensity mapped onto the surface indicates a low cost, and the brighter the intensity the higher the cost.

**Acknowledgments.** Calculations were performed at the University of California at Berkeley, the Lawrence Berkeley Laboratory, and the Technion, Israel Institute of Technology. This work was supported in part by the Applied Mathematics Subprogram of the Office of Energy Research under contract DE–AC03–76SF00098, and the National Science Foundation and Office of Naval Research under contract FDN00014-96-1-0381. RK is a Samuel & Miriam Wein Academic Lecturer, his research was partially supported by the fund for the promotion of research at the Technion.

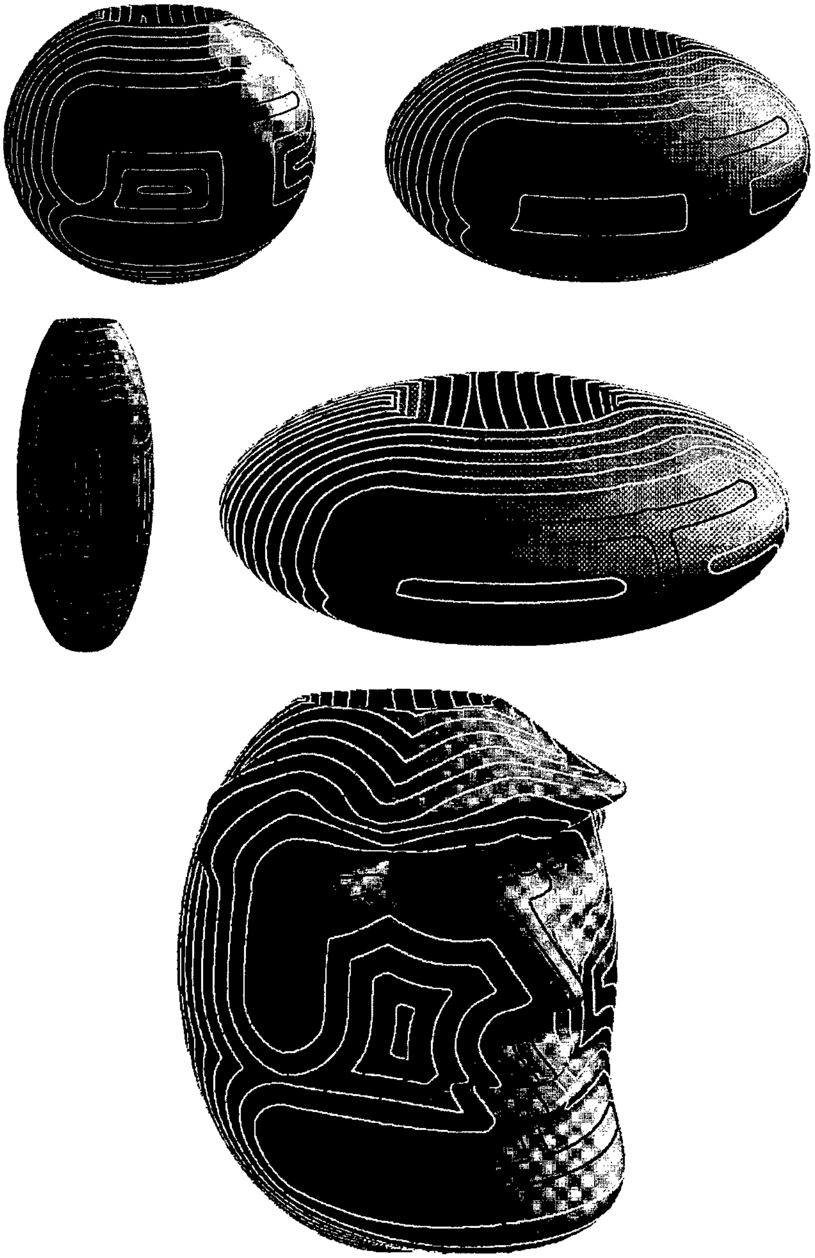

**Fig. 1.** Offsets on four beads and a Synthetic Head.

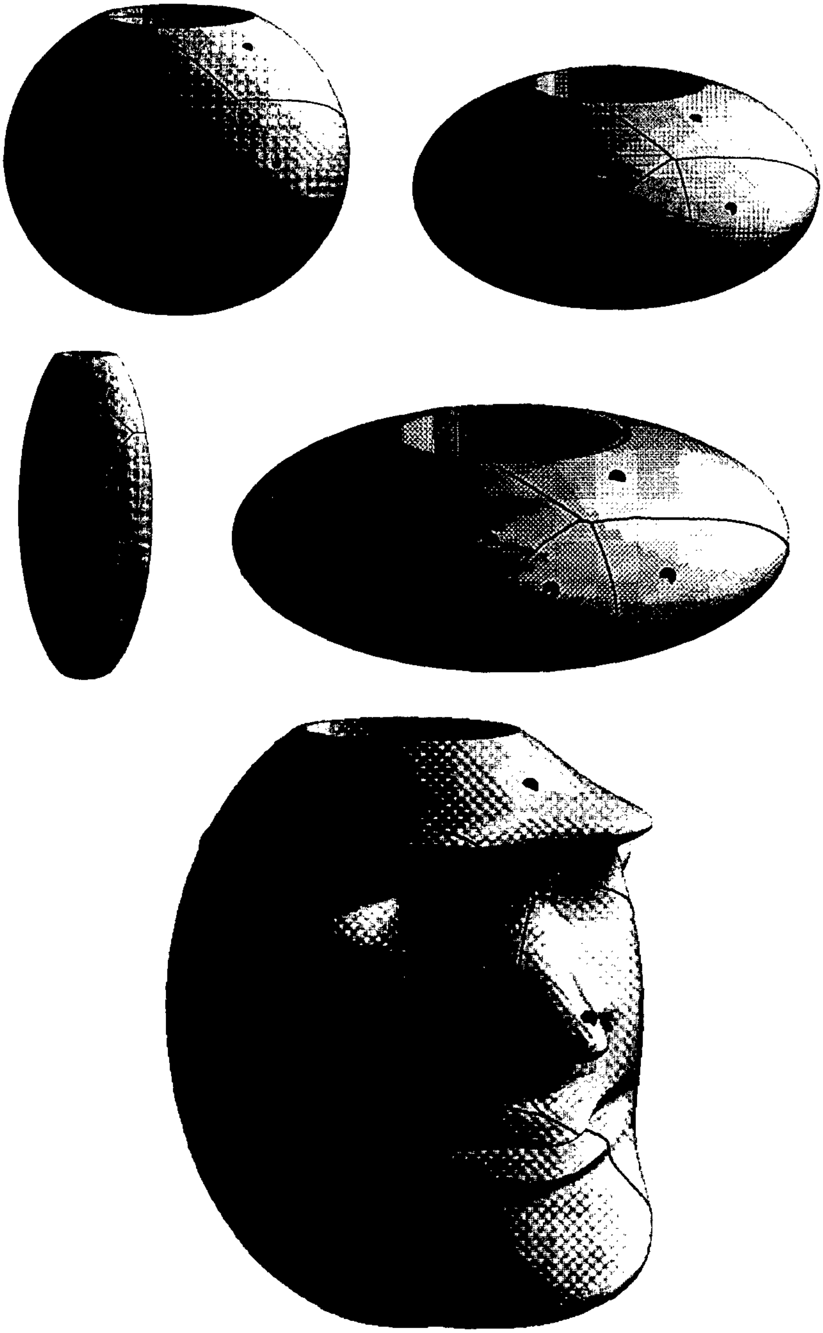

**Fig. 2.** Voronoi diagrams of five points on four beads and a Synthetic Head.

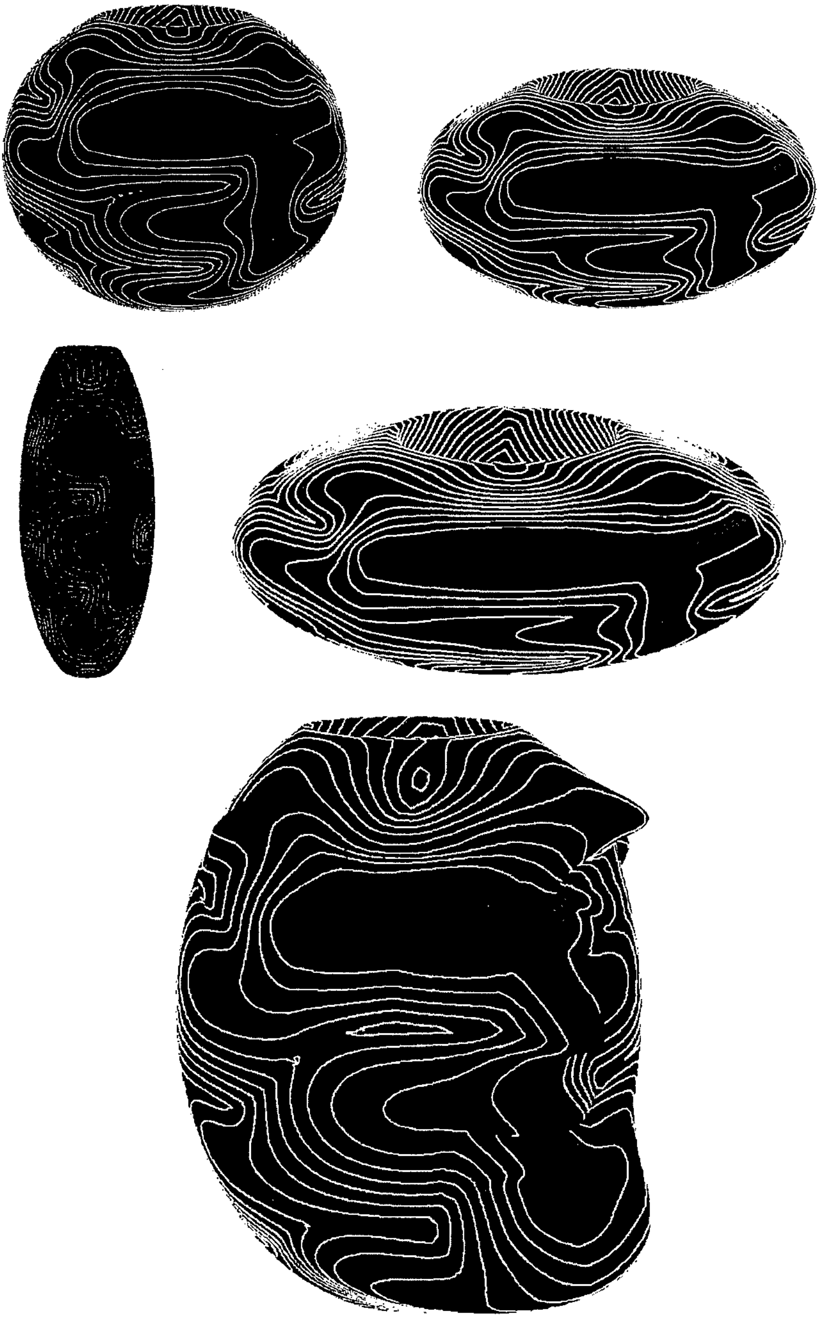

**Fig. 3.** Weighted offsets on four beads and a Synthetic Head.

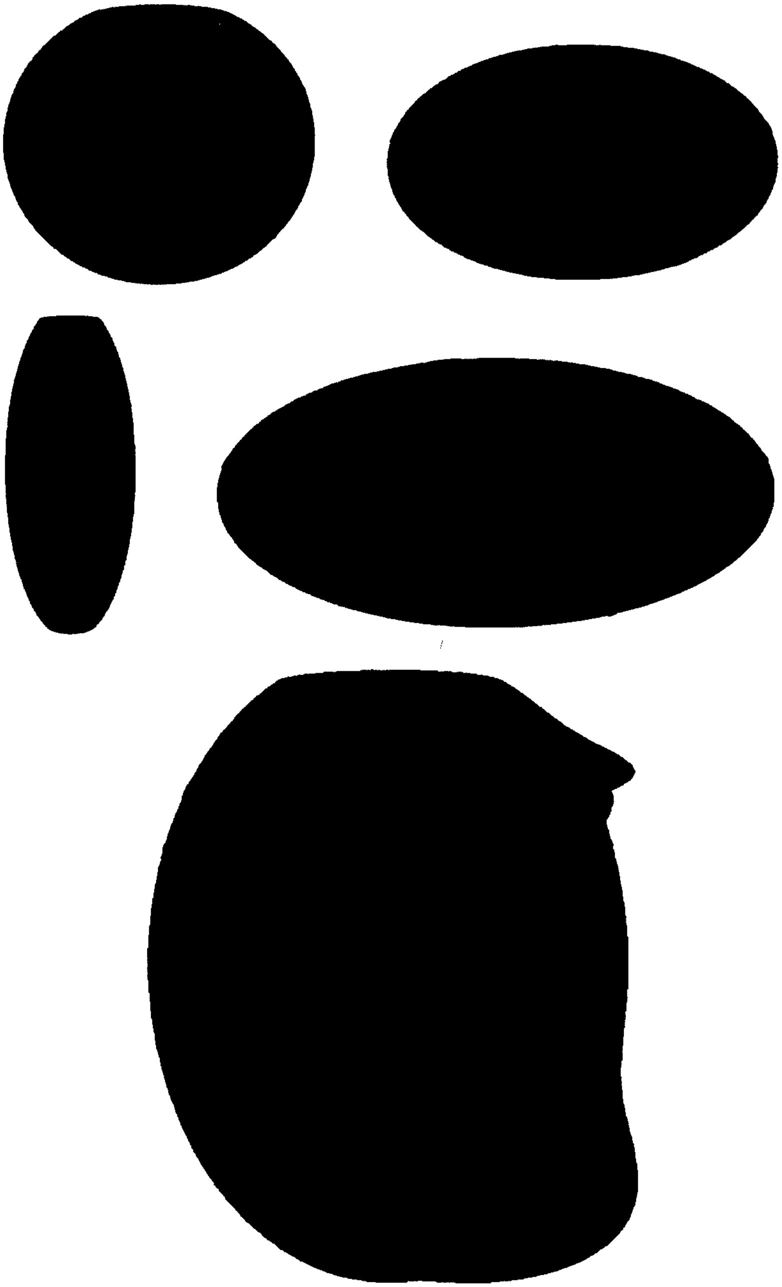

**Fig. 4.** Weighted Voronoi diagrams of five points on four beads and a Synthetic Head.

## References

1. Barth, T., and J. A. Sethian, Numerical schemes for the Hamilton-Jacobi and level set equations on triangulated domains, J. Comp. Phys. **145**(1) (1998), 1–40.

2. Bellman, R., and R. Kalaba, *Dynamic Programming and Modern Control Theory*, London mathematical society monographs, London, 1965.

3. Brockett, R. W., and P. Maragos, Evolution equations for continuous–scale morphology, in *Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing*, San Francisco, California, 1992, 1–4.

4. Cohen, L. D., and R. Kimmel, Global minimum for active contours models: A minimal path approach, International Journal of Computer Vision **24**(1) (1997), 57–78.

5. Dijkstra, E. W., A note on two problems in connection with graphs, Numerische Mathematic **1** (1959), 269–271.

6. Elber, G., and E. Cohen, Error bounded variable distance offset operator for free form curves and surfaces, Int. J. Comput. Geom. & Applic. **1**(1) (1991), 67–78.

7. Godunov, S. K., Finite difference method for numerical computation of discontinuous solution of the equations of fluid dynamics, Matematicheskii Sbornik **47** (1959), 271.

8. Kimmel, R., A. Amir, and A. M. Bruckstein, Finding shortest paths on surfaces using level sets propagation, IEEE Trans. Pattern Anal. and Machine Intelligence **17**(6) (1995), 635–640.

9. Kimmel, R., and A. M. Bruckstein, Shape offsets via level sets, Computer-Aided Design **25**(5) (1993), 154–162.

10. Kimmel, R., N. Kiryati, and A. M. Bruckstein, Sub-pixel distance maps and weighted distance transforms, Journal of Mathematical Imaging and Vision, Special Issue on Topology and Geometry in Computer Vision **6** (1996), 223–233.

11. Kimmel, R., and J. A. Sethian, Computing geodesic paths on manifolds, Proceedings of National Academy of Sciences, USA, **95**(15) (1998), 8431-8435.

12. Kunze, R., F. E. Wolter, and T. Rausch, Geodesic Voronoi diagrams on parametric surfaces, Welfen lab. report, Welfen Lab., Univ. Hannover, Germany, July 1997.

13. Maekawa, T., Computation of shortest paths on free-form parametric surfaces, Journal of Mechanical Design **118** (1996), 499–508.

14. Malladi, R., and J. A. Sethian, An O(N log N) algorithm for shape modeling, Proceedings of National Academy of Sciences, USA, **93** (1996), 9389–9392.

15. Mitchell, J. S. B., D. M. Mount, and C. H. Papadimitriou, The discrete geodesic problem, SIAM J. Comput. **16**(4) (1987), 647-668.

16. Osher, S. J., and J. A. Sethian, Fronts propagating with curvature dependent speed: Algorithms based on Hamilton-Jacobi formulations, J. of Comp. Phys. **79** (1988), 12–49.

17. Rouy, E., and A. Tourin, A viscosity solutions approach to shape-from-shading, SIAM. J. Numer. Analy. **29**(3) (1992), 867–884.

18. Sedgewick, R., *Algorithms*, Addison-Wesley, 1988.

19. Sethian, J. A., A review of the theory, algorithms, and applications of level set methods for propagating interfaces, Acta Numerica, Cambridge University Press, 1996.

20. Sethian, J. A., *Level Set Methods: Evolving Interfaces in Geometry, Fluid Mechanics, Computer Vision and Materials Sciences*, Cambridge Univ. Press, 1996.

21. Sethian, J. A., Curvature and the evolution of fronts, Comm. in Math. Phys. **101** (1985), 487–499.

22. Sethian, J. A. A marching level set method for monotonically advancing fronts, Proc. Nat. Acad. Sci. **93**(4), 1996.

Ron Kimmel
Computer Science Department
Technion, Israel Institute of Technology
Haifa 32000, Israel
ron@cs.technion.ac.il
http://www.cs.technion.ac.il/~ron/

James A. Sethian
Department of Mathematics
and Lawrence Berkeley National Laboratory
University of California, Berkeley, CA 94720, USA
sethian@math.berkeley.edu